

ES



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/387,927	09/01/1999	ANDY HARJANTO	1018.049US1	3178

41505 7590 08/25/2005  
WOODCOCK WASHBURN LLP  
ONE LIBERTY PLACE - 46TH FLOOR  
PHILADELPHIA, PA 19103

EXAMINER

LAO, SUE X

ART UNIT PAPER NUMBER

2194

DATE MAILED: 08/25/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

09/387,927

Applicant(s)

HARJANTO, ANDY

Examiner

Sue Lao

Art Unit

2194

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 21 June 2005.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-29 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-29 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).  
a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

### DETAILED ACTION

1. Claims 1-29 are pending. This action is in response to the amendments filed 2/8/2005 and 6/21/2005. Applicant has amended claims 1, 15, 18 and 24.
2. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.
3. Claim 27 is rejected under 35 U.S.C. 102(e) as being anticipated by Chan (US patent no. 5,893,107).

As to claim 27, Chan teaches  
directory (inherent throughout),  
at least one directory services (various directory services, col. 6 lines 7-13),  
directory services interface (OleDs container and leaf objects, col.6 lines 7-25)  
providing a common abstract interface (interface, col 7 lines 15-19),  
directory services interface extension (extending component, col. 5 lines 45-49  
and col. 6 lines 7 - 25 and col. 7 lines 15 - 19) providing an extended functionality  
(define new object classes and new properties).

4. Claims 1-13 and 15-26, 28, 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chan (US patent no. 5,893,107) in view of OLE 2 (OLE 2 Programmer's Reference, vol. 1, pages 36-39).

As to claim 1, Chan teaches the steps of:  
generating a desired directory extension (define new object classes and new properties, col. 5 lines 45-49 and col. 6 lines 7 - 25 and col. 7 lines 15 - 19);  
generating an interface software object (OleDs namespaces object / OLE/COM object, col. 4 line 35 - col. 5 line 18) to implement the desired directory extension using a predetermined software object framework (OLE/COM) having a class identification

Art Unit: 2194

(CLSID) and one or more interfaces (interfaces) each interface having an interface identification (IID of type REFIID);

inputting the generated interface software object (invoke function CoGetClassObject, col. 17 lines 4-9);

associating (binding, col. 9 line 6 - col. 10 line 8 / OleDs object corresponds to object of a directory service, col. 6 lines 7 - 25) one of a directory class (object class, col. 6 lines 17-19) and a directory attribute (object, col. 6 lines 15-17) to the class identification (CLSID, col. 4 line 49) of the interface software object (OleDs object) as stored in a predetermined location (registry, col. 4 line 48 - 50).

Chan teaches the interface software object provides an interface for accessing underlying directory services (each OleDs container object exposes an interface through which its contained objects can be accessed by a client, col. 7 lines 15-19). Such containing and exposing characteristics meet the typical definition of object aggregation, although Chan does not explicitly uses the term.

Nevertheless, OLE 2 teaches (section "Aggregation", pages 36-39) that making an object aggregatable or non-aggregatable is an implementation/design decision (aggregation is entirely optional and designing for it is an optional decision to be made for each object, page 38, first para. / page 36, first para. and last para.). Therefore, it would have been obvious to implement the interface software object of Chan as aggregatable.

One of ordinary skill in the art would have been motivated to combine the teaching of Chan and OLE 2 because the reward would have been larger than the cost in that it would have enabled the effective reuse of interface implementations (OLE 2, page 38, first para.).

As to claim 2, Chan teaches querying (CoGetClassObject, col. 17 lines 7-9), to expose an interface (class factory interface, lines 10-14).

As to claim 3, Chan teaches creating an instance (IClassFactory::CreateInstance, col. 17 line 12).

As to claim 4, Chan teaches invoking an interface (invokes IClassFactory, line 12) via interface identification (function returns pointer to a class factory interface, lines 10-11).

As to claim 5, Chan teaches creating an instance (create instance, col. 17 lines 11-16).

As to claim 6, Chan teaches  
creating (inherent) the object, assigning the class identification (CLSID, col. 4 lines 37-41);

creating and implementing one or more interfaces (inherent), assigning an interface identification (REFIID, inherent) for each interface.

As to claim 7, Chan teaches COM (COM, col. 4 line 60-67 / OLE 2.01, col. 4 lines 41-43).

As to claim 8, Chan teaches NT Directory Services (WinNTDS, col. 9 lines 55-57).

As to claims 9 and 10, see the rejection to claim 1 which meets the limitations of these claims.

As to claims 11 and 12, Chan teaches a client location (client program, inherently at client location, col. 4 lines 35-37), comprising a registry (registry, col. 4 line 48).

As to claim 13, Chan teaches a server location (inherent, OleDs maintains a registry, col. 7 lines 1-7).

As to claim 15, Chan teaches

Art Unit: 2194

querying (invoke function CoGetClassObject, col. 17 lines 4-9) a directory class (class for OleDs namespaces object) to expose the one or more interfaces (interface identifier, line 9) of an interface software object (OleDs namespaces object) having a class identification (class) previously associated (inherent);

invoking (invoke IClassFactory::CreateInstance, line 12) via interface identification (passing interface identifier, lines 12-13);

creating an instance (IClassFactory::CreateInstance method creates an instance, lines 10-16). Referring to claim 16, Chan teaches creating an instance (IClassFactory::Createinstance, col. 17 line 13) upon querying (CoGetClassObject, lines 7-8).

Chan further teaches a desired directory extension (define new object classes and new properties, col. 5 lines 45-49 and col. 6 lines 7 - 25 and col. 7 lines 15 - 19). Chan teaches the one or more interfaces of the interface software object (OleDs namespaces object / OLE/COM object, col. 4 line 35 - col. 5 line 18) implement a desired directory extension (each OleDs container object exposes an interface through which its contained objects can be accessed by a client, col. 7 lines 15-19). Such containing and exposing characteristics meet the typical definition of object aggregation, although Chan does not explicitly uses the term.

Nevertheless, OLE 2 teaches that making an object aggregatable or non-aggregatable is an obvious implementation/design decision (aggregation is entirely optional and designing for it is an optional decision to be made for each object, page 38, first para. / page 36, first para. and last para.). Therefore, it would have been obvious to implement the interface software object of Chan as aggregatable.

One of ordinary skill in the art would have been motivated to combine the teaching of Chan and OLE 2 because the reward would have been larger than the cost in that it would have enabled the effective reuse of interface implementations (OLE 2, page 38, first para.).

As to claim 16, see creating from the rejection to claim 3.

Art Unit: 2194

As to claim 17, see invoking and creating from the rejection to claim 15.

As to claims 18 - 23, and 24 - 26, see the rejection to the corresponding method claims 1 - 6, and 15 - 17 respectively.

As to claim 28, Chan as modified by OLE 2 teaches an aggregatable software object (OLE COM object, col. 4 lines 35-67) consistent with a predetermined software object framework (OLE / COM) and having a class identification (CLSID) and one or more interfaces (interfaces), each interface having an interface identification (interface identification I REFIID). See discussion of claim 1.

As to claim 29, Chan teaches the directory comprising a directory class (object class, col. 17-19) and a directory attribute (object, col. 6 lines 15-17), with the class identification stored in a permanent location (registry, lines 48-50).

5. Claim 14 is rejected under 35 U.S.C. 103(a) as being unpatentable over Chan in view of OLE 2 as applied to claim 13 and further in view of MSDN ("Lowering Total Cost of Ownership with Active DirectoryEnabled Applications").

As to claim 14, Chan as modified does not explicitly teach server location comprising a directory service.

MSDN teaches associations (names and locations of COM objects) are stored in a directory (directory tree, page 7).

It would be obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of Chan and MSDN's system because MSDN's COM storage location would be necessary for manipulating and maintaining the objects, and both references are related to using COM with directory services.

6. Applicant's arguments filed on 2/8/2005 have been considered but are moot in view of the new ground(s) of rejection.

Regarding the argued aggregatable (remarks, pages 9-10), this is explicitly met by OLE 2. Note discussion of claim 1 for detailed discussion. The combination of Chan and OLE 2 provided the aggregatable software object (aggregatable implementation of an interface software object).

Further, the passage of the specification cited by applicant (page 14, lines 13-17) reads as follows: "*The software object is what is known in the art as an aggregatable object. An aggregatable object, for example, in the context of COM, is an object that can be aggregated to another object. Aggregation is a specialized form of containment in the context of COM, for which COM provides special support. Aggregation allows an internal object's interfaces to be exposed as interfaces of an external object.*". Clearly as disclosed, the OLE aggregation is only one example of aggregation. OLE aggregation, internal object's interface and external object are not recited in the independent claims. See claims 1, 15, 18, 24 and 27.

7. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

8. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Sue Lao whose telephone number is (571) 272-3764. A voice mail service is also available at this number. The examiner's supervisor, SPE Meng-Ai An, can be reached on (571) 272 3756. The examiner can normally be reached on Monday - Friday, from 9AM to 5PM. The fax phone number for the organization where this application or proceeding is assigned is (703) 872 9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (571) 272-2100.



Art Unit: 2194

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

August 12, 2005



**SUE LAO**  
**PRIMARY EXAMINER**